

Approximate Message Passing with Unitary Transformation

Qinghua Guo and Jiangtao Xi

Abstract—Approximate message passing (AMP) and its variants, developed based on loopy belief propagation, are attractive for estimating a vector \mathbf{x} from a noisy version of $\mathbf{z} = \mathbf{Ax}$, which arises in many applications. For a large \mathbf{A} with i. i. d. elements, AMP can be characterized by the state evolution and exhibits fast convergence. However, it has been shown that, AMP may easily diverge for a generic \mathbf{A} . In this work, we develop a new variant of AMP based on a unitary transformation of the original model (hence the variant is called UT-AMP), where the unitary matrix is available for any matrix \mathbf{A} , e.g., the conjugate transpose of the left singular matrix of \mathbf{A} , or a normalized DFT (discrete Fourier transform) matrix for any circulant \mathbf{A} . We prove that, in the case of Gaussian priors, UT-AMP always converges for any matrix \mathbf{A} . It is observed that UT-AMP is much more robust than the original AMP for ‘difficult’ \mathbf{A} and exhibits fast convergence.

A special form of UT-AMP with a circulant \mathbf{A} was used in our previous work [13] for turbo equalization. This work extends it to a generic \mathbf{A} , and provides a theoretical investigation on the convergence.

Index Terms—Belief propagation, approximate message passing (AMP), convergence, singular value decomposition (SVD).

I. INTRODUCTION

PROXIMATE message passing, developed based on loopy belief propagation, is an efficient approach to the estimation of a vector \mathbf{x} with independent elements $\{x_i \sim p(x_i)\}$ in the following model

$$\mathbf{y} = \mathbf{Ax} + \mathbf{n} \quad (1)$$

where \mathbf{A} is a known matrix with size $M \times N$, the length of \mathbf{x} is N , \mathbf{y} denotes a length- M observation vector and \mathbf{n} is a length- M white Gaussian noise vector with zero mean and covariance matrix $\sigma^2 \mathbf{I}$ [1]-[7]. AMP was originally developed for compressive sensing based on model (1) [1]-[3], and then was extended to generalized AMP (GAMP) to accommodate more general distribution $p(y_i | (\mathbf{Ax})_i)$ which may not be Gaussian (where y_i and $(\mathbf{Ax})_i$ denotes the i -th element in \mathbf{y} and (\mathbf{Ax}) , respectively) [4], [5]. For a large \mathbf{A} with i.i.d. elements, AMP exhibits fast convergence which can be characterized by the state evolution [1], [5]. However, for a generic \mathbf{A} , the convergence of AMP cannot be guaranteed. It has been shown that AMP may diverge for a benign matrix \mathbf{A} and it can easily diverge for a ‘difficult’ matrix \mathbf{A} , e.g., non-zero mean, rank-deficient, column-correlated, or ill-conditioned \mathbf{A} [9], [12].

The fixed points and convergence of AMP were analyzed for an arbitrary matrix \mathbf{A} in [8] and [11]. Reference [8] provides sufficient conditions for the convergence of AMP in the case of

Gaussian priors $\{p(x_i)\}$. The convergence condition is closely related to the peak-to-average ratio of the squared singular values of a certain normalized \mathbf{A} for vector stepsize AMP algorithm, and is closely related to the peak-to-average ratio of the squared singular values of \mathbf{A} for scalar stepsize AMP algorithm. Damped AMP algorithms were proposed and the convergence can be guaranteed with sufficient damping, but the amount of damping grows with the peak-to-average ratio [8]. Adaptive damping and mean removal mechanisms were introduced to (G)AMP in [12] to enhance the convergence speed. Compared to original AMP, swept AMP (SwAMP) in [9], [10] is much more robust to difficult \mathbf{A} . However, SwAMP updates the relevant estimates sequentially (in contrast, AMP updates them in parallel), which restricts fast implementations. The global convergence of AMP with a generic \mathbf{A} for a generic prior $p(x_i)$ has not been understood [8].

In this work, we present a new variant of AMP, which is developed based on the following unitary transformation of (1)

$$\mathbf{r} = \mathbf{\Lambda} \mathbf{V} + \mathbf{w} \quad (2)$$

where $\mathbf{r} = \mathbf{U}^H \mathbf{y}$, $\mathbf{w} = \mathbf{U}^H \mathbf{n}$, and

$$\mathbf{A} = \mathbf{U} \mathbf{\Lambda} \mathbf{V} \quad (3)$$

with \mathbf{U} and \mathbf{V} being unitary matrices and $\mathbf{\Lambda}$ being a rectangular diagonal matrix. We note that, as \mathbf{U}^H is a unitary matrix, \mathbf{w} is still a zero mean Gaussian noise vector with the same covariance matrix as \mathbf{n} in (1). Eqn. (3) holds for any \mathbf{A} through the singular value decomposition (SVD). It is worth mentioning that, any circulant matrix \mathbf{A} ($M = N$) can be unitarily diagonalized by a discrete Fourier transform (DFT) matrix, so \mathbf{U} and \mathbf{V} can simply be the normalized DFT matrix and its inverse. In addition, \mathbf{r} and (the diagonal elements of) $\mathbf{\Lambda}$ can be calculated with the fast Fourier transform (FFT). A new variant of AMP is then developed based on model (2), which, for convenience, is called UT-AMP (where UT stands for unitary transformation) in this paper. It is interesting that, although unitary transformation does not change the singular values of \mathbf{A} , we prove that UT-AMP converges for any \mathbf{A} in the case of Gaussian priors. Moreover, we show that the convergence speed of UT-AMP is related to a scalar α (see Theorem 1 and its proof). It is observed that UT-AMP is much more robust than the original AMP algorithms and exhibits fast convergence. It is noted that the SVD required for a non-circulant \mathbf{A} only needs to be carried out once, so UT-AMP is particularly suitable for applications with a fixed \mathbf{A} (e.g., turbo MIMO detection with/without quasi-static channels in communications). For applications with model (1) with

The authors are with the School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, NSW, 2500, Australia, e-mail: {qguo,jiangtao}@uow.edu.au.

a circulant \mathbf{A} (e.g., block transmission with cyclic prefix in communications), the unitary transformation can be efficiently performed with FFT, which makes UT-AMP very attractive, e.g., in equalization to combat intersymbol interference, as shown in our previous work [13].

Notations: Bold lowercase letters, e.g., \mathbf{c} , are used to denote column vectors, and bold upper case letters, e.g., \mathbf{C} , are used to denote matrices. The i -th element in vector \mathbf{c} is denoted by c_i . We use $\mathbf{c} \cdot \mathbf{d}$ and \mathbf{c}/\mathbf{d} to denote the elementwise product and division between two vectors \mathbf{c} and \mathbf{d} , respectively. $|\mathbf{C}|^2$ represents the elementwise magnitude squared operation for matrix \mathbf{C} . $\mathbf{1}$, $\mathbf{0}$ and \mathbf{I} represent an all-one column vector, an all-zero column vector, and an identity matrix with proper sizes depending on the context. The conjugate transpose is denoted by the superscript “ H ”.

II. AMP WITH VECTOR STEPSIZES AND SCALAR STEPSIZES

To facilitate comparisons with UT-AMP, we include the vector stepsize AMP (Algorithm 1) and the scalar stepsize AMP (Algorithm 2) [5] in this section. In vector stepsize AMP, the function $g_x(\mathbf{q}, \tau_q)$ returns a column vector whose i -th element, denoted by $[g_x(\mathbf{q}, \tau_q)]_i$, is given by

$$[g_x(\mathbf{q}, \tau_q)]_i = \frac{\int x_i p(x_i) \mathcal{N}(x_i; q_i, \tau_{q_i}) dx_i}{\int p(x_i) \mathcal{N}(x_i; q_i, \tau_{q_i}) dx_i} \quad (4)$$

where $\mathcal{N}(x_i; q_i, \tau_{q_i})$ denotes a Gaussian distribution with x_i as random variable, q_i as mean, and τ_{q_i} as variance. Eqn. (4) can be interpreted as the MMSE (minimum mean square error) estimation of x_i based on the following model

$$q_i = x_i + \varpi \quad (5)$$

where $x_i \sim p(x_i)$ and ϖ is a Gaussian noise with mean zero and variance τ_{q_i} . The function $g'_x(\mathbf{q}, \tau_q)$ returns a column vector, and the i -th element is denoted by $[g'_x(\mathbf{q}, \tau_q)]_i$ where the derivative is with respect to q_i . It is not hard to show that $\tau_{q_i} [g'_x(\mathbf{q}, \tau_q)]_i$ is the a posteriori variance of x_i with model (5). Note that $g_x(\mathbf{q}, \tau_q)$ can also be changed for MAP (maximum a posteriori) estimation of \mathbf{x} [5].

Scalar stepsize AMP can be obtained from vector stepsize AMP by forcing the elements of each variance vector to be the same, so that the multiplications of a matrix with a vector in updating τ_p and τ_q are avoided (compare Lines 1 and 5 in both algorithms). The function $g_x(\mathbf{q}, \tau_q)$ is the same as $g_x(\mathbf{q}, \tau_q)$ in vector stepsize AMP except that all the Gaussian distributions $\{\mathcal{N}(x_i; q_i, \tau_q)\}$ share the same variance τ_q .

III. UT-AMP AND ITS CONVERGENCE

A. Derivation of UT-AMP

As any matrix \mathbf{A} can have the decomposition $\mathbf{A} = \mathbf{U}\Lambda\mathbf{V}$, we first perform a unitary transformation with \mathbf{U}^H to (1), yielding

$$\mathbf{r} = \mathbf{U}^H \mathbf{y} = (\mathbf{\Lambda}\mathbf{V})\mathbf{x} + \mathbf{w} \quad (6)$$

where $\mathbf{\Lambda}$ is an $M \times N$ rectangular diagonal matrix. Then the vector stepsize AMP can be applied to (6) where the system matrix becomes a special matrix $\mathbf{\Lambda}\mathbf{V}$. Note that

$$|\mathbf{C}|^2 \mathbf{d} = (\mathbf{C} \text{ Diag}(\mathbf{d}) \mathbf{C}^H) \mathbf{1} \quad (7)$$

Algorithm 1 Vector Stepsize AMP

Initialize $\tau_x^{(0)}$ (with elements larger than 0) and $\mathbf{x}^{(0)}$. Set $\mathbf{s}^{(-1)} = \mathbf{0}$ and $t = 0$.
Repeat

1. $\tau_p = |\mathbf{A}|^2 \tau_x^t$
2. $\mathbf{p} = \mathbf{Ax}^t - \tau_p \cdot \mathbf{s}^{t-1}$
3. $\tau_s = \mathbf{1} / (\tau_p + \sigma^2 \mathbf{1})$
4. $\mathbf{s}^t = \tau_s \cdot (\mathbf{y} - \mathbf{p})$
5. $1/\tau_q = |\mathbf{A}^H|^2 \tau_s$
6. $\mathbf{q} = \mathbf{x}^t + \tau_q \cdot \mathbf{A}^H \mathbf{s}^t$
7. $\tau_x^{t+1} = \tau_q \cdot g'_x(\mathbf{q}, \tau_q)$
8. $\mathbf{x}^{t+1} = g_x(\mathbf{q}, \tau_q)$
9. $t = t + 1$

Until terminated

Algorithm 2 Scalar Stepsize AMP

Initialize $\tau_x^{(0)} > 0$ and $\mathbf{x}^{(0)}$. Set $\mathbf{s}^{(-1)} = \mathbf{0}$ and $t = 0$.
Repeat

1. $\tau_p = (1/M) |\mathbf{A}_F|^2 \tau_x^t$
2. $\mathbf{p} = \mathbf{Ax}^t - \tau_p \mathbf{s}^{t-1}$
3. $\tau_s = 1 / (\tau_p + \sigma^2)$
4. $\mathbf{s}^t = \tau_s (\mathbf{y} - \mathbf{p})$
5. $1/\tau_q = (1/N) |\mathbf{A}^H_F|^2 \tau_s$
6. $\mathbf{q} = \mathbf{x}^t + \tau_q \mathbf{A}^H \mathbf{s}^t$
7. $\tau_x^{t+1} = (\tau_q/N) \mathbf{1}^H g'_x(\mathbf{q}, \tau_q)$
8. $\mathbf{x}^{t+1} = g_x(\mathbf{q}, \tau_q)$
9. $t = t + 1$

Until terminated

where $\text{Diag}(\mathbf{d})$ returns a diagonal matrix with the elements of \mathbf{d} on its diagonal, and $(\mathbf{B})_D$ returns a diagonal matrix by forcing the off-diagonal elements of \mathbf{B} to zero. Now suppose we have a variance vector τ_x^t . According to Line 1 in vector stepsize AMP and using (7), we have

$$\tau_p = (\mathbf{\Lambda}\mathbf{V} \text{ Diag}(\tau_x^t) \mathbf{V}^H \mathbf{\Lambda}^H)_D \mathbf{1}. \quad (8)$$

In attempting to reduce the computational complexity, we can find that if τ_x^t has a form of $\gamma \mathbf{1}$, the calculation of (8) can be significantly reduced. This motivates the replacement of τ_x^t

Algorithm 3 UT-AMP

Unitary transform: $\mathbf{r} = \mathbf{U}^H \mathbf{y} = \mathbf{\Lambda}\mathbf{V}\mathbf{x} + \mathbf{w}$, where $\mathbf{A} = \mathbf{U}\Lambda\mathbf{V}$.

Define vectors $\boldsymbol{\lambda}_p = \mathbf{\Lambda}\mathbf{\Lambda}^H \mathbf{1}$ and $\boldsymbol{\lambda}_s = \mathbf{\Lambda}^H \mathbf{\Lambda} \mathbf{1}$.

Initialize $\tau_x^{(0)} > 0$ and $\mathbf{x}^{(0)}$. Set $\mathbf{s}^{(-1)} = \mathbf{0}$ and $t = 0$.
Repeat

1. $\tau_p = \tau_x^t \boldsymbol{\lambda}_p$
 2. $\mathbf{p} = \mathbf{\Lambda}\mathbf{V}\mathbf{x}^t - \tau_p \cdot \mathbf{s}^{t-1}$
 3. $\tau_s = \mathbf{1} / (\tau_p + \sigma^2 \mathbf{1})$
 4. $\mathbf{s}^t = \tau_s \cdot (\mathbf{r} - \mathbf{p})$
 5. $1/\tau_q = (1/N) \boldsymbol{\lambda}_s^H \boldsymbol{\lambda}_s$
 6. $\mathbf{q} = \mathbf{x}^t + \tau_q (\mathbf{V}^H \mathbf{\Lambda}^H \mathbf{s}^t)$
 7. $\tau_x^{t+1} = (\tau_q/N) \mathbf{1}^H g'_x(\mathbf{q}, \tau_q)$
 8. $\mathbf{x}^{t+1} = g_x(\mathbf{q}, \tau_q)$
 9. $t = t + 1$
- Until terminated
-

with $\tau_x^t \mathbf{1}$ where τ_x^t is the average of the elements of $\boldsymbol{\tau}_x^t$. So (8) is reduced to

$$\boldsymbol{\tau}_p = \tau_x^t \boldsymbol{\Lambda} \boldsymbol{\Lambda}^H \mathbf{1} \quad (9)$$

which is Line 1 in UT-AMP. Lines 2, 3 and 4 in UT-AMP can be obtained according to Lines 2, 3, 4 in vector stepsize AMP by simply replacing \mathbf{A} with $\boldsymbol{\Lambda} \mathbf{V}$. According to (7) again, Line 5 in vector stepsize AMP with matrix $\boldsymbol{\Lambda} \mathbf{V}$ can be represented as

$$1./\tau_q = (\mathbf{V}^H \boldsymbol{\Lambda}^H \text{Diag}(\boldsymbol{\tau}_p) \boldsymbol{\Lambda} \mathbf{V})_D \mathbf{1}. \quad (10)$$

In order to reduce the computational complexity, we can replace the diagonal matrix $\boldsymbol{\Lambda}^H \text{Diag}(\boldsymbol{\tau}_p) \boldsymbol{\Lambda}$ in (10) with a scaled identity matrix $\rho \mathbf{I}$ where ρ is the average of the diagonal elements of $\boldsymbol{\Lambda}^H \text{Diag}(\boldsymbol{\tau}_p) \boldsymbol{\Lambda}$, i.e.,

$$\rho = (1/N) \mathbf{1}^H \boldsymbol{\Lambda}^H \boldsymbol{\Lambda} \boldsymbol{\tau}_p. \quad (11)$$

Hence (10) is reduced to Line 5 in UT-AMP. Line 6 can be obtained from Line 6 in vector stepsize AMP by replacing \mathbf{A} with $\boldsymbol{\Lambda} \mathbf{V}$. Compared with Line 7 in vector stepsize AMP, an additional average operation is performed in Line 7 in UT-AMP to meet the requirement of a scalar τ_x^t in Line 1. We note that the average operation is not necessarily in Line 7 as we can also put the additional average operation in Line 1. Line 8 in UT-AMP is the same as Line 8 in vector stepsize AMP except that τ_q is a scalar.

Remarks:

- One may try to get another variant of AMP by applying the scalar stepsize AMP to model (6), i.e., replacing \mathbf{A} with $\mathbf{U}^H \mathbf{A}$ and replacing \mathbf{y} with $\mathbf{r} = \mathbf{U}^H \mathbf{y}$ in scalar stepsize AMP. It is interesting that the obtained algorithm will remain exactly the same as the original scalar stepsize AMP as \mathbf{U}^H will be canceled out in scalar stepsize AMP. This means that unitary transformation have no impact on the convergence of scalar stepsize AMP.
- By the name, in vector stepsize AMP, $\boldsymbol{\tau}_x^t$, $\boldsymbol{\tau}_p$, $\boldsymbol{\tau}_s$, and $\boldsymbol{\tau}_q$ are all vectors, and in scalar step size AMP, the corresponding τ_x^t , τ_p , τ_s , and τ_q are all scalars. In contrast, UT-AMP has two scalars τ_x^t and τ_q and two vectors $\boldsymbol{\tau}_p$ and $\boldsymbol{\tau}_s$.
- If \mathbf{A} is a circulant matrix, UT-AMP is very attractive as \mathbf{U} and \mathbf{V} can be simply a DFT matrix and its inverse, and the diagonal elements of $\boldsymbol{\Lambda}$ can be calculated with FFT. Moreover, the multiplications of matrix and vector in UT-AMP can be implemented with FFT as well, leading to very low complexity.
- If \mathbf{A} is non-circulant and its SVD required in UT-AMP is available, the complexity per iteration of the UT-AMP is lower than that of vector stepsize AMP as the multiplications of matrix with vector are avoided in Lines 1 and 5. The complexity of UT-AMP is slightly higher than that of the scalar stepsize AMP due to the vector operations in Lines 1 and 5. Hence, UT-AMP is particularly suitable for applications with fixed \mathbf{A} as SVD only needs to be carried out once.
- Most importantly, it is observed that UT-AMP is robust to ‘difficult’ matrix \mathbf{A} and exhibits fast convergence.

B. Convergence of UT-AMP

Theorem 1. UT-AMP converges for any \mathbf{A} in the case of Gaussian priors. ■

Proof: See Appendix A.

It can be seen from the proof of Theorem 1 that, the convergence speed of UT-AMP is related to a parameter α given in (30) in Appendix A.

Similar to the original AMP, the convergence of UT-AMP for a generic prior is unknown, which remains as future work. It is also interesting to investigate the convergence of swept UT-AMP.

It is observed that UT-AMP is robust to ‘difficult’ \mathbf{A} e.g., non-zero mean, rank-deficient, column-correlated, or ill-conditioned \mathbf{A} , under which the original AMP often diverges. The special form of UT-AMP with a circulant \mathbf{A} in the case of discrete prior distributions has been used in [13] for equalization, where the channel matrix \mathbf{A} is ill conditioned, and the use of original AMP will diverge. Various numerical examples will be provided in the full version of this paper.

IV. CONCLUSION

In this work, we have developed a new AMP variant UT-AMP for a generic matrix \mathbf{A} . It has been shown that UT-AMP always converges in the case of Gaussian priors for any \mathbf{A} . It is observed that UT-AMP is robust to difficult \mathbf{A} and exhibits fast convergence.

APPENDIX A PROOF OF THEOREM 1

We assume that

$$\mathbf{p}(\mathbf{x}) \sim \mathcal{N}(\mathbf{x}; \mathbf{x}^0, \text{Diag}(\tau_x^0)) \quad (12)$$

where \mathbf{x}^0 and τ_x^0 are the a priori mean vector and variance vector for \mathbf{x} , respectively.

Similar to the proof in [8], it can be proven that the variance τ_x^t of UT-AMP for any \mathbf{A} always converges to a fixed point denoted by τ_x . Next, we prove the convergence of \mathbf{x}_t .

Define a diagonal matrix

$$\mathbf{D} = \text{Diag}(\boldsymbol{\tau}_s) = (\tau_x^t \boldsymbol{\Lambda} \boldsymbol{\Lambda}^H + \sigma^2 \mathbf{I})^{-1}. \quad (13)$$

Then, according to the UT-AMP algorithm, we have

$$\mathbf{s}^t = \tau_x^t \mathbf{D} \boldsymbol{\Lambda} \boldsymbol{\Lambda}^H \mathbf{s}^{t-1} - \mathbf{D} \boldsymbol{\Lambda} \mathbf{V} \mathbf{x}^t + \mathbf{D} \mathbf{r}, \quad (14)$$

$$\begin{aligned} 1/\tau_q &= (1/N) \mathbf{1}^H \boldsymbol{\Lambda}^H \mathbf{D} \boldsymbol{\Lambda} \mathbf{1} \\ &= \frac{1}{N} \sum_{i=1}^{\min\{M,N\}} \frac{|\lambda_i|^2}{\tau_x^t |\lambda_i|^2 + \sigma^2}, \end{aligned} \quad (15)$$

with λ_i being the (i, i) -th elements of $\boldsymbol{\Lambda}$. and

$$\begin{aligned} \mathbf{x}^{t+1} &= \tau_x^{t+1} (\mathbf{q}/\tau_q + \mathbf{x}^0 ./ \tau_x^0) \\ &= (\tau_x^{t+1}/\tau_q) \mathbf{x}^t + \tau_x^{t+1} \mathbf{V}^H \boldsymbol{\Lambda}^H \mathbf{s}^t + \tau_x^{t+1} \mathbf{x}^0 ./ \tau_x^0 \\ &= \tau_x^{t+1} \tau_x^t \mathbf{V}^H \boldsymbol{\Lambda}^H \mathbf{D} \boldsymbol{\Lambda} \boldsymbol{\Lambda}^H \mathbf{s}^{t-1} \\ &\quad + (\alpha \mathbf{I} - \tau_x^{t+1} \mathbf{V}^H \boldsymbol{\Lambda}^H \mathbf{D} \boldsymbol{\Lambda} \mathbf{V}) \mathbf{x}^t + \mathbf{b}, \end{aligned} \quad (16)$$

where \mathbf{b} is an appropriate vector. Define

$$\tilde{\alpha} = \frac{\tau_x^{t+1}}{\tau_q} = \frac{1}{N} \sum_{i=1}^{\min\{M,N\}} \frac{\tau_x^{t+1} |\lambda_i|^2}{\tau_x^t |\lambda_i|^2 + \sigma^2}. \quad (17)$$

The iteration of \mathbf{x}^t and \mathbf{s}^t in UT-AMP can be described as

$$\begin{bmatrix} \mathbf{s}^t \\ \mathbf{x}^{t+1} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{C}_a & \mathbf{C}_b \\ \mathbf{C}_c & \mathbf{C}_d \end{bmatrix}}_{\mathbf{C}} \begin{bmatrix} \mathbf{s}^{t-1} \\ \mathbf{x}^t \end{bmatrix} + \mathbf{e} \quad (18)$$

where \mathbf{e} is an appropriate vector. Matrix \mathbf{C} has two diagonal sub-matrices

$$\mathbf{C}_a = \tau_x^t \mathbf{D} \mathbf{\Lambda} \mathbf{\Lambda}^H, \quad (19)$$

and

$$\mathbf{C}_b = -\mathbf{D} \mathbf{\Lambda} \mathbf{V}, \quad (20)$$

and the other two sub-matrices can be represented as

$$\mathbf{C}_c = \tau_x^{t+1} \tau_x^t \mathbf{V}^H \mathbf{\Lambda}^H \mathbf{D} \mathbf{\Lambda} \mathbf{\Lambda}^H, \quad (21)$$

and

$$\mathbf{C}_d = \tilde{\alpha} \mathbf{I} - \tau_x^{t+1} \mathbf{V}^H \mathbf{\Lambda}^H \mathbf{D} \mathbf{\Lambda} \mathbf{V}. \quad (22)$$

Next, we find the eigenvalues of matrix \mathbf{C} , i.e., the roots of the following polynomial

$$h(\eta) = |\eta \mathbf{I} - \mathbf{C}| = \begin{vmatrix} \eta \mathbf{I} - \mathbf{C}_a & \mathbf{C}_b \\ \mathbf{C}_c & \eta \mathbf{I} - \mathbf{C}_d \end{vmatrix} = 0. \quad (23)$$

We note that the identity matrices in (23) have different sizes (i.e., the use of \mathbf{I} is abused for notation simplification). As \mathbf{C}_a is a diagonal matrix (with non-negative elements), a diagonal matrix $\omega \mathbf{I}$ can be used to guarantee that $\eta \mathbf{I} - \mathbf{C}_a + \omega \mathbf{I}$ is invertible. Define a new polynomial

$$h^a(\eta) = \begin{vmatrix} \eta \mathbf{I} - \mathbf{C}_a + \omega \mathbf{I} & \mathbf{C}_b \\ \mathbf{C}_c & \eta \mathbf{I} - \mathbf{C}_d \end{vmatrix}. \quad (24)$$

Clearly the roots of $h^a(\eta)$ with $\omega = 0$ are the eigenvalues of matrix \mathbf{C} . It can be shown that $h^a(\eta)$ can be rewritten as

$$\begin{aligned} h^a(\eta) &= |\eta \mathbf{I} - \mathbf{C}_a + \omega \mathbf{I}| \times |\eta \mathbf{I} - \mathbf{C}_d - \mathbf{C}_c (\eta \mathbf{I} - \mathbf{C}_a + \omega \mathbf{I})^{-1} \mathbf{C}_b| \\ &= |\eta \mathbf{I} - \tau_x^t \mathbf{D} \mathbf{\Lambda} \mathbf{\Lambda}^H + \omega \mathbf{I}| \times \\ &\quad |\mathbf{V}^H| \times |(\eta - \tilde{\alpha}) \mathbf{I} + \tau_x^{t+1} \mathbf{\Lambda}^H \mathbf{D} \mathbf{\Lambda} + \tau_x^{t+1} \tau_x^t \mathbf{\Lambda}^H \mathbf{D} \mathbf{\Lambda} \mathbf{\Lambda}^H \\ &\quad (\eta \mathbf{I} - \tau_x^t \mathbf{D} \mathbf{\Lambda} \mathbf{\Lambda}^H + \omega \mathbf{I})^{-1} \mathbf{D} \mathbf{\Lambda}| \times |\mathbf{V}|. \end{aligned} \quad (25)$$

As \mathbf{V} is a unitary matrix, $|\mathbf{V}^H| = |\mathbf{V}| = 1$. So they can be removed from (25). Note that $\mathbf{\Lambda}$ is a rectangular diagonal matrix with size $M \times N$ and all the matrices left in (25) are diagonal.

Case 1: $M = N$. In this case, $\mathbf{\Lambda}$ is a diagonal matrix. Define vector $\beta = [\beta_1, \dots, \beta_N]^T$ whose elements are the diagonal elements of $\tau_x^t \mathbf{D} \mathbf{\Lambda} \mathbf{\Lambda}^H$, i.e.,

$$\beta_i = \frac{\tau_x^t |\lambda_i|^2}{\tau_x^t |\lambda_i|^2 + \sigma^2}. \quad (26)$$

where we can see that $0 \leq \beta_i < 1$. Hence (25) can be rewritten as

$$\begin{aligned} h^a(\eta) &= \prod_{i=1}^N (\eta - \beta_i + w) \left((\eta - \tilde{\alpha}) + \frac{\tau_x^{t+1}}{\tau_x^t} \left(\beta_i + \frac{\beta_i^2}{\eta - \beta_i + w} \right) \right) \\ &= \prod_{i=1}^N (\eta - \beta_i + w) \left((\eta - \tilde{\alpha}) + \frac{\tau_x^{t+1}}{\tau_x^t} \beta_i \right) + \frac{\tau_x^{t+1}}{\tau_x^t} \beta_i^2. \end{aligned} \quad (27)$$

Letting $\omega = 0$, we have

$$h(\eta) = \prod_{i=1}^N (\eta - \beta_i) \left((\eta - \tilde{\alpha}) + \frac{\tau_x^{t+1}}{\tau_x^t} \beta_i \right) + \frac{\tau_x^{t+1}}{\tau_x^t} \beta_i^2. \quad (28)$$

As the variance always converges, we have $\tau_x^{t+1} = \tau_x^t = \tau_x$ after a certain number of iterations. Then (28) can be reduced to

$$h(\eta) = \prod_{i=1}^N (\eta^2 - \alpha \eta + \alpha \beta_i) \quad (29)$$

where from (17)

$$\alpha = \frac{1}{N} \sum_{i=1}^{\min\{M,N\}} \frac{\tau_x |\lambda_i|^2}{\tau_x |\lambda_i|^2 + \sigma^2}. \quad (30)$$

which indicates that $0 < \alpha < 1$. Hence the eigenvalues are given by

$$\eta_{i(1,2)} = \frac{-\alpha \pm \sqrt{\alpha^2 - 4\alpha\beta_i}}{2} \quad (31)$$

for $i = 1, 2, \dots, N$. Recall that $0 \leq \beta_i < 1$. If $\alpha \geq \beta_i/4$, the eigenvalues are real, and it can be easily shown that

$$|\eta_i| \leq \alpha < 1, \quad (32)$$

where the equality holds when $\beta_i = 0$. If $\alpha < \beta_i/4$, the eigenvalues are complex valued, and it can be shown that

$$|\eta_i| \leq \alpha \beta_i < 1. \quad (33)$$

Case 2: $M > N$. In this case, $\mathbf{\Lambda}$ is a ‘tall’ rectangular diagonal matrix. We define diagonal matrix $\tilde{\mathbf{\Lambda}}$ with size $N \times N$ as the upper part of $\mathbf{\Lambda}$, and define diagonal matrix $\tilde{\mathbf{D}}$ with size $N \times N$ as the upper left part of \mathbf{D} (whose size is $M \times M$). It is not hard to show that

$$\begin{aligned} h^a(\eta) &= |\eta \mathbf{I}_{M-N} + \omega \mathbf{I}_{M-N}| \times |\eta \mathbf{I} - \tau_x^t \tilde{\mathbf{D}} \tilde{\mathbf{\Lambda}} \tilde{\mathbf{\Lambda}}^H + \omega \mathbf{I}| \\ &\quad \times |(\eta - \alpha) \mathbf{I} + \tau_x^{t+1} \tilde{\mathbf{\Lambda}}^H \tilde{\mathbf{D}} \tilde{\mathbf{\Lambda}} + \tau_x^{t+1} \tau_x^t \tilde{\mathbf{\Lambda}}^H \tilde{\mathbf{D}} \tilde{\mathbf{\Lambda}} \tilde{\mathbf{\Lambda}}^H \\ &\quad (\eta \mathbf{I} - \tau_x^t \tilde{\mathbf{D}} \tilde{\mathbf{\Lambda}} \tilde{\mathbf{\Lambda}}^H + \omega \mathbf{I})^{-1} \tilde{\mathbf{D}} \tilde{\mathbf{\Lambda}}|. \end{aligned} \quad (34)$$

After some manipulations, we have

$$h(\eta) = \eta^{M-N} \prod_{i=1}^N (\eta^2 - \alpha \eta + \alpha \beta_i). \quad (35)$$

Hence the eigenvalues are the same as those in Case 1 except that $M - N$ eigenvalues are zero.

Case 3: $M < N$. In this case, $\mathbf{\Lambda}$ is a ‘fat’ rectangular diagonal matrix. Define diagonal matrix $\bar{\mathbf{\Lambda}}$ with size $M \times M$ as the left part of $\mathbf{\Lambda}$. We can show that

$$\begin{aligned} h^a(\eta) &= |\eta \mathbf{I} - \tau_x^t \bar{\mathbf{D}} \bar{\mathbf{\Lambda}} \bar{\mathbf{\Lambda}}^H + \omega \mathbf{I}| \times |(\eta - \alpha) \mathbf{I} + \tau_x^{t+1} \bar{\mathbf{\Lambda}}^H \bar{\mathbf{D}} \bar{\mathbf{\Lambda}} + \\ &\quad \tau_x^{t+1} \tau_x^t \bar{\mathbf{\Lambda}}^H \bar{\mathbf{D}} \bar{\mathbf{\Lambda}} \bar{\mathbf{\Lambda}}^H (\eta \mathbf{I} - \tau_x^t \bar{\mathbf{D}} \bar{\mathbf{\Lambda}} \bar{\mathbf{\Lambda}}^H + \omega \mathbf{I})^{-1} \bar{\mathbf{D}} \bar{\mathbf{\Lambda}}| \\ &\quad \times |(\eta - \alpha) \mathbf{I}_{N-M}|. \end{aligned} \quad (36)$$

Then, we can have

$$h(\eta) = (\eta - \alpha)^{N-M} \prod_{i=1}^M (\eta^2 - \alpha \eta + \alpha \beta_i). \quad (37)$$

The eigenvalues are the same as those in Case 1 except that $N - M$ eigenvalues are α .

The above shows that $|\eta_i| \leq \alpha$ for all the cases (noting that $\alpha \beta_i < \alpha$) and any matrix \mathbf{A} . Because α is smaller than 1, the algorithm converges for any \mathbf{A} .

REFERENCES

- [1] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Natl. Acad. Sci.*, vol. 106, no. 45, pp. 18914-18919, 2009.
- [2] D. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing: I. motivation and construction," in *IEEE Information Theory Workshop (ITW)*, 2010, pp. 1 -5.
- [3] D. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing II:analysis and validation," in *Proc. Info. Theory Workshop*, Jan. 2010.
- [4] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. Inform. Theory*, Saint Petersburg, Russia, Jul.-Aug. 2011, pp. 2174-2178.
- [5] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," arXiv:1010.5141v1 [cs.IT], Oct. 2010.
- [6] S. Rangan, "Estimation with random linear mixing, belief propagation and compressed sensing," in *Information Sciences and Systems (CISS)*, 2010 44th Annual Conference on. IEEE, 2010, pp. 1-6.
- [7] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inform. Theory*, vol. 57, no. 2, pp. 764-785, Feb. 2011.
- [8] S. Rangan, P. Schniter, and A. Fletcher, "On the convergence of generalized approximate message passing with arbitrary matrices," in *Proc. IEEE Int. Symp. Inform. Thy.*, pp. 236-240, July 2014. (full version at arXiv:1402.3210).
- [9] F. Caltagirone, F. Krzakala, and L. Zdeborova, "On convergence of approximate message passing," *arXiv preprint arXiv:1401.6384*, Jan. 2014.
- [10] A. Manoel, F. Krzakala, E. W. Tramel, and L. Zdeborova, "Sparse estimation with the swept approximated message-passing algorithm," arXiv:1406.4311, Jun. 2014.
- [11] S. Rangan, P. Schniter, E. Riegler, A. Fletcher, and V. Cevher, "Fixed points of generalized approximate message passing with arbitrary matrices," in *Proc. ISIT*, Jul. 2013, pp. 664-668.
- [12] J. Vila, P. Schniter, S. Rangan, F. Krzakala, and L. Zdeborova, Adaptive damping and mean removal for the generalized approximate message passing algorithm, in *IEEE ICASSP*, 2015, submitted.
- [13] Q. Guo, D. Huang, S. Nordholm, J. Xi, and Y. Yu, "Iterative Frequency Domain Equalization With Generalized Approximate Message Passing" *Signal Processing Letters, IEEE* , vol.20, no.6, pp.559-562, Jun. 2013.